

Управление на файлове и директории

ас. Стоян Мечев

катедра „Информационни технологии“

ВВМУ „Н. Й. Вапцаров“

Файлове

- Определено количество информация записано върху постоянен носител и на което е дадено име.
- Правила за имената на файловете:
 - На теория имената на файловете могат да съдържат всеки знак, който се извежда на екран с изключение на нулевия байт (ASCII 0) и слеш "/";
 - Използването на знаци като \$, интервал или умлаути изисква знак за "избягване" за да не се изтълкуват като част от команда;
 - Linux и linux са два различни файла;
 - Препоръчва се използването на стандартните символи a-z A-Z 0-9 +-. _ от гледна точка на съвместимост с различните дистрибуции.
 - Не се използват разширения.
 - Програмата `convtn` преобразува имена на файлове от една кодова таблица в друга.
 - Дължината на името може да бъде до 255 знака (за старите Unix системи до 14 знака). Винаги първият знак е буква, цифра или "." .
 - Валидни имена: `my_file`; `my.file.01`; `01.myfile`; `.myfile`;
 - Невалидни имена: `-50-shades`; `3/4_littres`; `straße-nöe`.
 - **Внимание!** Файловете с имена започващи с "." се скриват: ***.my_file***, но системата работи с тях.

Директории

- Използват се за да групират файловете в йерархична структура. Особен вид файлове.
 - Контролира достъпа до файловете;
 - Важат същите правила за именуване;
 - Основна директория "/" (root, да не се бърка с потребителя root);
 - Уникалност на имената на файловете в директорията.
- Абсолютни и относителни пътища.
 - Абсолютен път – започва от "/". /bin/lib/
 - Относителен път – започва от текущата директория "." ../../bin/
 - . – текуща директория;
 - .. – една директория над текущата.
 - Текущата директория се наследява между процесите – например при стартиране на нов шел.
- \$PATH
 - при стартиране на файл
 - ./script (когато file не е включен в \$PATH)
- Домашна директория
 - Обозначава се със знака "~"
 - /home/user_name
 - обикновено е същото име, като на потребителя.

ОСНОВНИ КОМАНДИ

- **cd** – сменя директория
 - само **cd** ни прехвърля в домашната директория на потребителя;
 - "**cd -**" прехвърля ни в директорията, която преди това е била текуща;
- **pwd** – показва текущата директория
- **cp** – копира файл
- **mv** – премества или преименува файл
- **rm** – изтрива файл **-rf**
- **mkdir** – създава директория
- **rmdir** – изтрива директория
- **ls** – показва списъка на файловете в директория

ls – параметри

- a --all показва всички файлове, включително скритите;
- i --inode показва inode на файловете;
- l --format=long показва подробна информация за файловете;
- o --no-color не оцветява различните типове файлове;
- r или -F показва типа на файла, като го маркира със специален знак
"/" директория; "*" изпълним файл; "|" FIFO файл; "@" линк; "=" сокет.
- r --reverse сортира в обратен ред;
- R --recursive преминава през поддиректориите;
- S --sort=size сортира по размер;
- t --sort=time сортира по време на промяна;
- X --sort=extension сортира по разширение;
- d когато имаме нужда от информация за самата директория, като такава.

файлове

- `ls -la`
- *inode*
(запис за всеки файл)
- Тип
- Права - собственик, група, останалите
- Hardlinks
- Собственик
- Група
- Дължина в байтове
- Дата на последна промяна
- Име

```
ian@Z61t-u14:~/lpi103-2$ ls -al
```

```
total 52
```

drwxrwxr-x	2	ian	ian	4096	Jun	8	17:09	.
drwxr-xr-x	15	ian	ian	4096	Jun	8	13:26	..
-rw-rw-r--	1	ian	ian	8	Jun	8	17:02	sedtab
-rw-rw-r--	1	ian	ian	24	Jun	8	13:26	text1
-rw-rw-r--	1	ian	ian	25	Jun	8	13:36	text2
-rw-rw-r--	1	ian	ian	63	Jun	8	16:19	text3
-rw-rw-r--	1	ian	ian	26	Jun	8	16:19	text4
-rw-rw-r--	1	ian	ian	24	Jun	8	16:42	text5
-rw-rw-r--	1	ian	ian	98	Jun	8	17:09	text6
-rw-rw-r--	1	ian	ian	15	Jun	8	15:48	xaa
-rw-rw-r--	1	ian	ian	9	Jun	8	15:48	xab

mkdir, rmdir

mkdir dir1_name dir2_name създава две директории;

mkdir -p dir1_name/dir2_name създава вложени директории;

rmdir изтрива директория, ако е празна;

rmdir -p изтрива всички вложени директории, ако са празни;

Шаблони за имена на файлове

- За търсене на скрити файлове е необходимо изрично да се укаже "."
- ? – заменя един знак
 - ls ?? .txt
- * заменя нула или повече знаци
 - ls a*.txt
 - Внимание! ls *, ако съществува файл, който се казва '-l'
- [] – клас на замяна – заменя един символ
 - [0-9a-f] – от 0 до 9 включително и от a до f включително
 - [!0-9] – всеки знак с изключение на 0 до 9
 - ls [123]file.txt
- Къдрави скоби {}
 - {red, yellow, blue}.txt, се заменят от шел с red.txt, yellow.txt, blue.txt;
 - {a,b}{1,2}.txt = a1.txt, a2.txt, b1.txt, b2.txt;
 - няма значение, дали файловете съществуват.

cp, mv, rm

- **cp** копира файл

- cp file1 file2 - копира съдържанието на file1 във file2
- cp file1 my_dir - копира file1 в директорията my_dir
- приема шаблони;
- полезни опции:
 - i интерактивно - пита дали да презаписва файлове, които съществуват;
 - R рекурсивно - копира файловете от поддиректориите;
 - u копира файла само ако изходният файл е по-нов от целевия;
 - f презаписва съществуващи файлове, без да пита;
 - v дава подробна информация за действията;

- **mv** мести файл. Ползва същия синтаксис като cp.

- използва се за преименуване на файлове;

- **rm** изтрива файл. Също не задава въпроси при изтриването на файловете. Да се ползва с голямо внимание, особено **rm -rf** и при шаблони!

- **rm -r ***
- **rm -i** по-приемлив вариант;

more, less, find

- **more** и **less** се използват за интерактивен преглед на съдържанието на файлове.
 - **find** търси рекурсивно по определени критерии
 - name приема шаблон за търсене;
 - type задава типа на файла: f обикновен файл, d директория, l линк и др.;
 - user указва, кой е собственикът на файла;
 - group към коя група принадлежи файла;
 - size определя размер на търсеният файл. +10k всички файлове с размер над 10k;
 - atime време на достъпване на файла;
 - mtime време на промяна на файла;
 - ctime време на промяна на атрибутите на файла;
 - perm определя разрешения за достъп, които трябва да има търсеният файл;
 - inum намира линковете към файл със съответния inode;
 - Ако са зададени няколко условия, то по подразбиране между тях се изпълнява логическо "и". Добре е да се използват () за да се избегне объркване.
 - Възможно е ! - логическо "не" или -o "или"
 - Пример за логическо "или" **find . -name file3 -o -name file4**
 - Може да се задават допълнителни действия, като стартира отделен процес за всеки файл:
 - `find . -user ivan -exec ls -l '{}' \;`
 - `find . -atime +13 -exec rm -i '{}' \;`
 - за да не се създава отделен процес за всеки намерен файл може да се използва командата xargs:
`find . -atime +13 | xargs -r rm -i`
- Внимание!** Има особености - например, ако имената на файловете съдържат интервал.

Търсене на файлове - примери

- `find . -name "*.htm" -ok mv '{}' '{}l' \;`
 - преименува всички файлове от *.htm на *.html
 - '{}' - заменя името на файла
 - \; - край на командата
- `find . -empty -exec rm '{}' \;` - изтрива всички празни файлове
- `find . -iwholename "ACK/*1"` - търси включително в пътя
- `find . -type d` (d - directory, f - file, l - link)
- `find . -size 0` - търси празни файлове
- `find . -size -26c -size +23c -print` (c - байтове, k - килобайтове)

Линкове ln и ln -s

- няколко имена на един файл:
 - ln target linkname
 - ls -i
- За удобство и защита:
 - изтриването на линка не изтрива файла;
- „Твърдите“ линкове не се създават към директории;
- Към директории сочат симлинковете;
 - могат да сочат извън файловата система;
 - целта „не знае“ за тях;
 - повредени симлинкове.

Бързо търсене на файлове

- Командата **find** е много мощна, но може да се изпълнява много бавно.
- **locate**
 - търси в база данни;
 - когато търси само стринг, то го търси, където и да се намира в името на файла;
 - когато се ползва шаблон за търсене, намира само файловете, които отговарят напълно на шаблона;
 - **updatedb** – обновява базата на locate;
 - **Внимание!** Базата **locate** съдържа различна информация, в зависимост от потребителя, който изпълнява командата **updatedb**;
 - `--localpaths='path1 path2...'`
 - знаците "/" и "." не се обработват по специален начин;
- **slocate** търси в собствена подробна база данни
 - нейната база поддържа информация за правата за достъп до файловете и показва само тези, които съответния потребител има право да види.

- Благодаря за вниманието!
- Въпроси?